

# 1 Introduction and Project Overview

Is it possible to realize value from distributed data without conflicting with security and privacy concerns? Specifically, can organizations optimize allocation of global resources while preserving the privacy of local information? The focus of this proposal is to develop privacy-preserving collaborative optimization techniques that would allow organizations to gain the maximum value from local information without (or with limited) information disclosure. While answering this problem, an inherent aim of the proposal is to also solve some of the fundamental problems underlying privacy-preserving analysis / secure computation and make it more accessible and applicable.

Optimization is a fundamental problem found in many diverse fields. Research in optimization methods has generated many successes; the ubiquitous collection of data opens even greater opportunities. Much of this data is constrained by privacy and security concerns, preventing the sharing and centralization of data needed to apply optimization techniques. This proposal tackles the challenge of developing privacy-preserving distributed optimization techniques. This would greatly increase the value of collected data and prevent data warehouses from turning into data cemeteries.

Innumerable such situations exist, where appropriate safe and secure use of data leads to immense financial and social benefits. For example, consider the case of Ford and Firestone. In 2001, numerous accidents due to tread separation were reported. Initially both companies blamed each other. It turned out that it was only Ford Explorers with Firestone tires from the Decatur, Illinois plant, in specific situations that had these problems. If found out earlier, much loss could have been avoided. While both companies individually collect a lot of pertinent testing data, this was not shared due to commercial concerns. In the packaged goods industry, delivery trucks are empty 25% of the time. Just two years ago, Land O'Lakes truckers spent much of their time shuttling empty trucks down slow-moving highways, wasting several million dollars annually. By using a web based collaborative logistics service (Nistevo.com), to merge loads from different companies (even competitors) bound to the same destination, huge savings were realized (freight costs were cut by 15%, for an annual savings of \$2 million[80]). Though this required sending all information to a central site, there was no alternative if savings were desired. However, the proposed project *would* make this possible without the release of proprietary information. Finally, Walmart, Target and CostCo ship millions of dollars worth of goods over the seas every month. These feed in to their local ground transportation network. The cost of sending half-empty ships is prohibitive, but the individual corporations have serious problems with disclosing freight information. If it were possible simply to determine what trucks should make their way to which ports to be loaded onto certain ships i.e., solve the classic transportation problem, without knowing the individual constraints, the savings would be enormous. In all of these cases, complete sharing of data would lead to invaluable savings/benefits. However, since unrestricted data sharing is a competitive impossibility or requires great trust, better solutions must be found.

Solutions have been devised for similar data privacy problems. Data is often anonymized and released for public use. For example, census data is transformed to release micro-samples. However, this brings the problem of privacy into sharp focus. Our personal data is *supposed* to be private. Nevertheless, as several high profile infractions have shown, this is not really the case. Recent research [78] also shows that it is still possible to breach the privacy of such anonymized data. An alternative is to anonymize the data with respect to some metric such as  $k$ -anonymity or  $l$ -diversity. While a lot of effort has been spent on considering the security aspects of such solutions, very little effort has been spent on measuring the utility of such transformed data. It is unclear whether such approaches can be directly adapted for secure optimization. Another approach comes from cryptography – Secure Multi-party Computation (SMC) is a very powerful formalism developed in the cryptographic community that sets a standard for what it means to provably maintain privacy and security. Advances in this field show that it is possible to securely compute any function over distributed data. However, they also do not completely solve the problem. The developed methods are quite costly in terms of the computational cost. They also do not address the problem of what is learned through the results of the computation. Much of the work relies on simplistic semi-honest model – mechanism design can ensure that all parties stay true to the protocol. However not all functions are amenable to this. A last possibility is to only use local data. This obviates all concerns about security and privacy, but is not practically useful. Therefore, the key challenge is how to use data without really having complete access to it in a safe and privacy-preserving manner. Solving this problem would have tremendous societal impact since the applications of this technology are multitudinous.

The proposed project will advance this state of the art by developing a suite of distributed privacy-

preserving optimization techniques using secure computation tools and methodologies. In particular, we will develop privacy-preserving variants of several specific optimization techniques as well as more general solutions. This will involve solving several fundamental problems such as finding new definitions that are more relaxed than the standard SMC definitions, yet still accurately model the real security concerns. This is a very challenging research task due to the subtle nature of security and security definitions. Optimization modeling also requires significant time. The general procedure that can be used in the process cycle of modeling is to: (1) mathematically formulate the problem, (2) find an optimal solution, and (3) control the problem by assessing/updating the optimal solution continuously, while changing the parameters and structure of the problem. Clearly, there are always feedback loops among these general steps. However, this means that several methods may need to be applied, and the problem parameters changed and solutions recomputed as required. This adds to the security problem. Even if a single method does not disclose too much information, combining multiple methods or even multiple iterations of the same method might cause privacy leaks or inference problems. Rather than measuring privacy loss in terms of a single technique, we need more general metrics that apply over the broad spectrum of usable life of the data. An implicit objective is to evaluate existing metrics by this criteria, and propose others, if necessary.

It is also important to evaluate what would actually be necessary to enable strategic information sharing across organizations. To an extent, the notion of secure computation is completely unknown in this problem domain. The idea of secure computation has very little penetration in the real world. The goal of this research is to explore problems where data is distributed, evaluate the value to be gained by analysis over the distributed data, and to propose secure solutions for such analysis. Purely technical solutions are often insufficient. Thus, one of the research foci of this project will be to determine what are the managerial constraints and requirements to enable such information sharing. This may include creating and distributing a survey among CIOs to create awareness as well as gauge response. Overall, successfully enabling secure information sharing would create a strategic advantage for corporations, and empower organizations to realize the true value of collected data.

## 1.1 Overview of the Proposed Activities

This project addresses optimization tasks with the following four assumptions: 1) data is distributed across multiple sources, and security/privacy concerns limit the sharing of data (in particular, preventing the centralization of data), 2) if the data were centralized, optimization techniques would be used to identify the optimal (or good) operating points, 3) optimization should be possible without needing a privacy-preserving method for each and every different optimization technique, and 4) utilizing multiple methods should not increase privacy loss (or at the very least, the privacy implications should be well understood). This project will develop techniques of securely transforming the data in some form, or develop broad based secure techniques for utilization. Rather than measuring privacy loss in terms of a single technique, we need metrics that apply over the broad spectrum of usable life of the data. An implicit objective is to evaluate existing metrics by this criteria, and propose others, if necessary.

Initially, we will focus on Linear Programming, since it is one of the most important subfields in optimization, applicable to numerous real problems. A secure solution for linear programming would be sufficient to solve the transportation problem discussed earlier. To have maximum impact, any study must start with Linear Programming. However, many interesting optimization problems are nonlinear. Quadratic Programming comprises an area of optimization whose broad range of applicability is second only to linear programs. In general, Nonlinear optimization provides fundamental insights into mathematical analysis and is widely used in a variety of fields such as engineering design, regression analysis, inventory control, geophysical exploration, and economics. Finally, constraint satisfaction is one of the most important problems in AI and industrial decision problems. Thus, to focus our effort, we aim to develop efficient algorithms for secure optimization in the following areas: (1) linear programming, (2) quadratic programming, (3) specific flavors of nonlinear programming, and (4) constraint satisfaction.

To be useful in practice, it is not enough to show that these problems are theoretically efficient (i.e., polynomial by the  $O(\cdot)$  notation). The vast secure computation literature already proves the feasibility of this and provides a generic method. Moreover, the field of optimization abounds with methods, that though theoretically correct, are quite unusable in practice. This might be due to sensitivity to bad data, or numerical computation problems or other reasons. Thus, in order to be *really* useful and worthwhile, any developed method has to be implemented and tested on real data. We will continue developing secure techniques that

have proven to be successful in our prior work on privacy-preserving data mining [86, 82, 81, 38]. Overall, we plan to build research prototypes of tools for performing secure optimization and evaluate their effectiveness and usability. Even though we choose specific fields in optimization as our primary problem domains, we expect that insights and methodologies obtained in this context will be useful in many other settings as well. The fact that secure computation techniques are applicable in many different settings, supports this expectation.

The education activities in this project encompass course development and student mentoring both at the undergraduate and graduate levels. A key objective is to introduce MBA students to the basic concepts of secure computation. This should foster penetration into the real world. The PI will develop a graduate-level course on secure optimization and information sharing and disseminate the course materials through a dedicated website. The website will also be used to maintain resources for research on secure computation and optimization. Results from the proposed research will be closely integrated into the course. The PI will also make special efforts to participate in faculty development programs aimed at increasing the number of Information Assurance and Security (IAS) professionals educated by US colleges and universities. For example, the Information Assurance Education Graduate Certificate Program at Purdue, funded by the National Security Agency, brought together many educators from different institutes together. While this is now over, such opportunities are ongoing, and the PI will make special efforts to attend and contribute to such efforts. This would enable the PI to disseminate the research results and course materials developed in this project broadly to faculty members at many different institutions. The PI is also collaborating with faculty members at minority-serving institutions in their efforts to develop an IAS curriculum.

## 1.2 Roadmap

The rest of the proposal is structured as follows. A survey of work related to the proposal is given in Section 2. Section 3 provides the research plan. Here, we present the problem domain, discuss global challenges and research tasks, and present preliminary results. The education plan is described in Section 4. We then present the timeline for carrying out the research and education activities in Section 5, and discuss the metrics for evaluating success in Section 6.

## 2 Related Work

Privacy is an important problem especially relevant to current times. Several research communities have independently developed solutions which contribute to privacy protection. Much of this work can be leveraged, if appropriately combined. We first discuss work in the field of secure multi-party computation. We also discuss some application of this to work in the data mining community. We then overview relevant work from the field of distributed optimization.

### 2.1 State of the art in Secure Computation

The concept of Secure Multi-party Computation (SMC) is especially relevant to this proposal. The setting of SMC encompasses tasks as simple as coin-tossing and broadcast, and as complex as electronic voting, electronic auctions, electronic cash schemes, contract signing, anonymous transactions, and private information retrieval schemes. The key idea behind Secure Multi-party Computation is that a computation is secure if at the end of the computation, no party learns anything except its own input and the results (and anything that can be inferred from these two pieces). The gold standard is that of a trusted third party which performs the entire computation. Thus, the key is to achieve the same results without having a trusted third party. While some communication is obviously required in order to perform the computation, it is necessary to stick to messages that are useful yet do not reveal anything new. How is this possible? The answer lies in non-determinism. By allowing non-determinism in the exact values sent in the intermediate communication (e.g., encrypt with a randomly chosen key), and proving that a party using its own input and the result can generate a predicted intermediate computation that is as likely as the actual values is sufficient to show that no new information is revealed.

Secure computation has a very rich history. Yao first postulated the two-party comparison problem (Yao's Millionaire Protocol) and developed a provably secure solution [89]. Goldreich et al. [34] generalized this to multi-party computation and proved that there exists a secure solution for any functionality. The approach used is as follows: the function  $F$  to be computed is first represented as a combinatorial circuit, and then the

parties run a short protocol for every gate in the circuit. Every participant gets random shares of the input and output wires for every gate. This approach, though appealing in its generality and simplicity, means that the number of rounds of the protocol grow with the size of the circuit. This grows with the size of the input. This is highly inefficient for large inputs or complicated circuits, as in optimization. Although this proves secure solutions exist, achieving efficient secure solutions for distributed optimization is still open.

There has been significant theoretical work in this area. Both [89] and [34] assumed polynomially time bounded passive adversaries. In a line of work initiated by Ben-Or et al.[10], the computational restrictions on the adversary were removed, but users were assumed to be able to communicate in pairs in perfect secrecy. Ben-Or et al.[10] assume passive adversaries, while Chaum et al.[19] extend this to active adversaries. Ostrovsky and Yung [60] introduce the notion of mobile adversaries, where the corrupt users may change from round to round. Finally, the coercing adversary who can force users to choose their inputs in a way he favors was introduced in the context of electronic elections by Benaloh and Tuninstra[12], and generalized to arbitrary multi-party computation by Canetti and Gennaro[15]. Much effort has been devoted to developing crisp definitions of security [56, 35, 16]. However, due to efficiency reasons, it is completely infeasible to directly apply the theoretical work from SMC to form secure protocols for optimization.

Secure Multi-party Computation does make two key contributions to the proposed work: 1. Methods for securely computing functions with small inputs (e.g., secure comparison), and 2. Definitions and proof techniques for private and secure computations in a distributed environment. Creating efficient secure variants of different optimization techniques is a non-trivial task requiring much effort. Rather than using SMC directly to build a new solution for every optimization problem, a much better solution is to use SMC tools to build secure generalized methods to transform the data in a way that other techniques could then be applied.

### 2.1.1 State of the art in Privacy-preserving data analysis

Along with electronic auctions[30, 13, 59, 14], another field that has gained a lot of attention in recent years is Privacy-preserving data mining. Privacy-preserving data mining is really the application of secure computation to data mining. Work in PPDM has followed two major directions – the randomization/perturbation approach and the cryptographic approach. Both of these are of some interest to us.

In the perturbation approach data is locally perturbed by adding “noise” before mining is done. For example, if we add a random number chosen from a gaussian distribution to the real data value, the data miner no longer knows the exact value. However, important statistics on the collection (e.g., average) will be preserved. Special techniques are used to reconstruct the original distribution (not the actual data values). The mining algorithm is modified to work while taking this into consideration. The seminal paper by Agrawal and Srikant [2] introduced this notion to the data mining community – univariate additive noise from either a uniform or gaussian distribution is used to locally perturb the data. Several different algorithms are proposed to reconstruct distributions and learn a decision tree classifier from the perturbed data. A convergence result was proved in [1] for a refinement of this algorithm. There has been work using the same approach for association rule mining [28, 71, 95]. Zhu and Lei[97] study the problem of optimal randomization for privacy-preserving data mining and demonstrate the construction of optimal randomization schemes for density estimation. This model works in the “data warehouse” model of data mining, but trades off privacy for accuracy of results. However, several problems have been pointed out with the privacy inherent in such an approach[57, 41, 36].

The second approach to PPDM is a direct application of secure computation techniques to the data mining problem. Typically, cryptographic techniques are used to protect privacy. Lindell and Pinkas[51, 52] first proposed this to construct decision trees. There has since been significant work on many techniques in data mining. Specifically, secure methods have been proposed for association rule mining [86, 40], clustering[47, 82, 37], classification [27, 83, 85, 88, 93, 94], outlier detection [84] and regression[42, 72]. Kantarcioglu and Vaidya[38] propose an architecture for the mining of information. Clifton et al.[21] propose the creation of a toolkit of components that could be used to quickly throw together a solution for a new data mining problem. The underlying tools can actually be used for any problem. Thus, some of the techniques and tools developed in this field are equally applicable to the field of optimization, and the results found by this project would be applicable to all other applications of SMC as well.

## 2.2 State of the art in Optimization methods

Optimization problems occur in all walks of real life. There is work in distributed optimization that aims to achieve a global objective using only local information. This falls in the general area of distributed decision making with incomplete information. This line of research that has been investigated in a worst case setting (with no communication between the distributed agents) by Papadimitriou et al. [61, 22, 62]. In [62], Papadimitriou and Yannakakis first explore the problem facing a set of decision-makers who must select values for the variables of a linear program, when only parts of the matrix are available to them and prove lower bounds on the optimality of distributed algorithms having no communication. Awerbuch and Azar[6] propose a distributed flow control algorithm with a global objective which gives a logarithmic approximation ratio and runs in a polylogarithmic number of rounds. Bartal et al.’s distributed algorithm [8] obtains a better approximation while using the same number of rounds of local communication.

Distributed Constraint Satisfaction was formalized by Yokoo[90] to solve naturally distributed constraint satisfaction problems. These problems are divided between agents, who then have to communicate among themselves to solve them. To address distributed optimization, complete algorithms like OptAPO and ADOPT have been recently introduced. ADOPT[58] is a backtracking based bound propagation mechanism. It operates completely decentralized, and asynchronously. The downside is that it may require a very large number of messages, thus producing big communication overheads. OptAPO[53] centralizes parts of the problem; it is unknown apriori how much needs to be centralized where, and privacy is an issue. Distributed local search methods like DSA([43]) / DBA([96]) for optimization, and DBA for satisfaction ([91]) start with a random assignment, and then gradually improve it. Sometimes they produce good results with a small effort. However, they offer no guarantees on the quality of the solution, which can be arbitrarily far from the optimum. Termination is only clear for satisfaction problems, and only if a solution was found.

DPOP[67] is a dynamic programming based algorithm that generates a linear number of messages. However, in case the problems have high induced width, the messages generated in the high-width areas of the problem become too large. There have been proposed a number of variations of this algorithm that address this problem and other issues, offering various tradeoffs (see [68, 64, 63, 66, 65, 69]). [63] proposes an approximate version of this algorithm, which allows the desired tradeoff between solution quality and computational complexity. This makes it suitable for very large, distributed problems, where the propagations may take a long time to complete.

However, in general, the work in distributed optimization has concentrated on reducing communication costs and has paid little or no attention to security constraints. Thus, some of the summaries may reveal significant information. In particular, the rigor of security proofs has not been applied much in this area. There is some work in secure optimization. Silaghi and Rajeshirke[75] show that a secure combinatorial problem solver must necessarily pick the result randomly among optimal solutions to be really secure. Silaghi and Mitra[73] propose arithmetic circuits for solving constraint optimization problems that are exponential in the number of variables for any constraint graph. A significantly more efficient optimization protocol specialized on generalized Vickrey auctions and based on dynamic programming is proposed by Suzuki and Yokoo[77], though it is not completely secure under the framework in [75]. Yokoo et al.[92] also propose a scheme using public key encryption for secure distributed constraint satisfaction. Silaghi et al.[74] show how to construct an arithmetic circuit with the complexity properties of DFS-based variable elimination, and that finds a random optimal solution for any constraint optimization problem. Atallah et al.[5] propose protocols for secure supply chain management. However, much of this work is still based on generic solutions and not quite ready for practical use. Even so, some of this work can definitely be leveraged to advance the state of the art by building general transformations or privacy-preserving variants of well known methods.

## 3 Research Plan

Optimization is the study of problems in which one seeks to minimize or maximize a real function by systematically choosing the values of real or integer variables from within an allowed set. Formally, given a function  $f : A \rightarrow R$  from some set  $A$  to the real numbers, we seek an element  $x_0$  in  $A$  such that  $f(x_0) \leq f(x), \forall x \in A$  (“minimization”) or such that  $f(x_0) \geq f(x), \forall x \in A$  (“maximization”). Thus, an optimization problem has three basic ingredients:

- An *objective function* which we want to minimize or maximize.
- A set of *unknowns* or *variables* which affect the value of the objective function.

- A set of *constraints* that allow the unknowns to take on certain values or exclude others.

Optimization problems have been well studied in the literature – methods have been proposed for the case when all of the data is available at a central site. Methods have also been proposed for the case with incomplete information (distributed optimization). However all solution methods assume that all the necessary data is centralized or freely available. Privacy/security causes a problem whenever the data is distributed. There are many ways in which data could be distributed. Each of the ingredients of the optimization problem could be distributed. For example, the objective function might be known to only one party, or parts of it known to some subsets of the parties. The constraints that define the set  $A$  might also be distributed in some fashion. Different parties might own different constraints or even different parts of the same constraint. Thus, the different ways in which data is distributed give rise to the following categorization:

**Horizontal Partitioning / Homogeneous Distribution:** Here, each constraint would be fully owned by one party. Thus, different parties own different constraints. An example of this would be the distributed scheduling problem. Suppose that several schedulers need to schedule tasks on machines. Each task can be executed by several machines (though not all), and it can be split between several machines, but the fraction of all tasks executed by a machine must under no circumstances exceed its capacity. Each scheduler only knows the tasks that may be executed on its pertinent machines, and based on this information it must decide what fractions of its task to send to which machines. The sum of all fractions is to be maximized. Here, the objective function could be known to a single party or to all of the parties, or even be shared by the parties.

**Vertical Partitioning / Heterogeneous Distribution:** In this case, each constraint is shared between some subset of the parties. An example of this would be the organization theory problem. A large enterprise has a very extensive set of tasks – say, products manufactured. A fundamental question in Organization Theory is, *how are these tasks to be partitioned among managers?* Although the profitability and resource requirements of these products may change dynamically with market conditions, the *constraint structure*, the sparsity pattern of the constraint matrix of the associated linear program, may be fixed. That is, it is known in advance, which products compete for which resources. What are the *organizational principles* that should guide this assignment of tasks to managers, so that the latter can make more informed decisions. Again, the objective function might be known to all of the parties, or just to a single party, or be shared by the parties.

**Arbitrary Partitioning:** Apart from the prior two partitioning methods, the data may also be arbitrary partitioned in some way (some combination of the above). This is more general and subsumes both of the earlier cases. Completely arbitrary partitioning of data is unlikely in practice, though certain specific configurations might easily be found. In any case, solutions for this case will always work for both of the prior cases as well.

We now look at possible solution approaches to the problem. The effectiveness of any solution can be measured on the basis of three critical properties – privacy/security, utility, and efficiency. Does the solution preserve the privacy/security of the data? How useful is the solution (i.e., how accurate are the analysis results)? How efficient is the solution? Any solution that completely disregards even one of these properties is not of any use, while the ideal solution would perfectly satisfy all three parameters. However, most current solutions focus on meeting one, largely satisfying the second, and pay lip service to the third. Instead, we would like to develop solutions that largely satisfy at least two parameters, and at the same time take the third parameter into consideration to a certain minimum critical level.

There are three basic solution approaches to the privacy-preserving distributed collaborative analysis problem: 1) data transformation 2) synthetic data creation and 3) secure computation. Each of the three are different in terms of what problems they can solve as well as the privacy/utility tradeoff. With data transformation, the data is transformed in some manner and then analysis algorithms are applied over it. The complete global transformation may not be known or will be non-invertible. Metrics such as  $k$ -anonymity and  $l$ -diversity place constraints on the space of acceptable transformations to ensure the security of the data. However, the utility of the transformed data is still open to question. Do we still get accurate results? In general, the transformation must be carefully chosen balancing concerns of privacy/security and utility.

With synthetic data creation, instead of the actual data, equivalent data is created in some random way. This is instead used for analysis. In the context of optimization, this could translate into creating equivalent constraints and/or an equivalent objective function. Any constraints/objective function leading to the same

final optimal solution is acceptable. However, again, what does the process of creating the new data leak and how can data comparable to the global constraints be created?

The third solution approach is to build a secure protocol to directly solve the problem. This could be done using the generic techniques developed in SMC or by developing more efficient solutions using other cryptographic primitives. One problem with this is that a secure protocol must be built for every algorithm that needs to be applied over the data. This makes it difficult to use for exploratory activities. The best solution might be to create a hybrid approach by combining the prior approaches. We can create a secure protocol for an appropriate data transformation method that would allow us to apply different solvers to solve the problem. At the same time, we can also create secure versions of specific important distributed as well as centralized optimization algorithms to create a good tool kit of applicable methods. We now briefly discuss the problem domains and then lay out the fundamental research challenges inherent, before giving a sample of the representative solutions.

### 3.1 Problem Domains

By itself, Optimization is an entire subject of study and is subdivided into several major fields of research that are important in their own right. We focus specifically on Linear Programming and certain subfields of Nonlinear programming. Linear programming studies the case in which the objective function  $f$  is linear and the set  $A$  is specified using only linear equalities and inequalities. Integer programming studies linear programs in which some or all variables are constrained to take on integer values. Quadratic programming allows the objective function to have quadratic terms, while the set  $A$  must be specified with linear equalities and inequalities. Nonlinear programming studies the general case in which the objective function or the constraints or both contain nonlinear parts. Constraint satisfaction studies the case in which the objective function  $f$  is constant (this is used in artificial intelligence, particularly in automated reasoning). Together these form an excellent set of problem domains that are broad, challenging, and have tremendous impact.

There are many alternative optimization algorithms available. It is important to recognize the characteristics of a problem and identify the appropriate solution technique. Within each class of problems, there are different minimization methods, which vary in computational requirements, convergence properties, and so on. Each of these sub-fields is immense in scope, and there are well over 4000 solution algorithms for different kinds of optimization problems. A brute force approach would be to create secure solutions for all of them. However, we would like to focus specific important problems, create leading solutions and spark widespread interest in the field. Apart from this, we would like to gain key insights to the general problem. It should be unnecessary to create 4000 different secure programs. Instead, we should be able to formulate efficient generic solutions to particular sub-fields. The aim is also to collaborate with leading researchers in each domain, introduce the notion of security and multi-party computation, and leverage joint skills. Thus, we will focus research on Linear Programming as well as Quadratic Programming to begin with, since these are two of the most important sub fields in optimization, and Rutgers has world renowned expertise in the area in the MSIS department as well as in RUTCOR (Rutgers Center for Operations Research). This will be followed by constraint satisfaction (since there is already prior work in this area that can be leveraged) as well as Nonlinear Programming. While creating secure solutions for specific problems, on a parallel track work will proceed on solving some of the fundamental problems underlying secure computation. We now discuss these.

### 3.2 Fundamental Research Challenges

We now look at some of the fundamental challenges underlying privacy-preserving collaborative computation that must be met to make it accessible and practical. These apply to privacy-preserving optimization as well.

#### 3.2.1 Utility metrics for Transformation

There are known techniques for transforming the dataset into an equivalent privacy-compliant dataset. However, the key question is to what extent do such transformations retain the utility of data? We need utility metrics that accurately measure how useful the data remains. One possibility is to use degree of difference – i.e., the amount of difference between the original and the transformed data. However, this may not accurately reflect the utility of the data. For example, consider two transformations – one in which all of the constraints are slightly changed, while another where certain constraints are significantly changed. While the total difference might be greater in the second transformation, those changes may be redundant and the

optimal solution may remain the same, while in the first case the optimal solution may be changed. The problem is that the difference that matters is with the optimal solution (i.e., in general according to a specific metric/goal) which is typically unknown before solving the problem. Instead we will explore alternatives including sensitivity analysis to help figure out how well the transformation works. Sensitivity analysis can help to gauge the degree of tolerance the problem has towards approximate data transformation. In itself, the tolerance level of different data sets to transformation may be different.

### 3.2.2 Modeling Exterior Knowledge and Result Analysis

Exterior Knowledge needs to be effectively modeled to decide whether a particular computation may lead to breach of privacy or not. The theory of secure multiparty computation is only concerned with the correct and secure evaluation of a function – not about what exterior knowledge exists and how it affects the privacy/security. However, this has to be accounted for for real and practical use. For optimization, exterior knowledge could itself be modeled in the form of constraints on the problem. One possibility is to write the constraints on the private data and then run a constraint satisfaction algorithm to find feasible answers. By modeling exterior knowledge also in the form of constraints, we can judge the difference in availability of feasible solutions and thus determine the degree of effect the exterior knowledge has. Similarly, result analysis is increasingly crucial to practical secure computation. Even if the function is securely computed, there is still a problem even if the results themselves leak information. [39] show one approach to quantifying this. We will further extend this to apply in other situations as well.

### 3.2.3 Iteration

Iterative algorithms (viz., iteration) is a significant obstacle to efficient secure computation. Many optimization techniques require iteration in some form or fashion. The effect of iteration on security is enormous. It is possible to write a secure algorithm such that the results of each iteration are also kept secret, but this is certain to lead to severe problems with efficiency. Instead, each iteration could probably be independently made secure much more efficiently. The question then is what to do with the intermediate results? Can they be protected in some form? What do they reveal? What is the threshold for security – how many iterations are to be allowed before a security violation occurs. Indeed, is it possible to define the parameters for a secure violation? The issue of composability of secure protocols as well as parallelizability has been well studied[50, 18, 17, 49, 44, 48]. But the issue of iteration needs to be similarly addressed.

### 3.2.4 Integral values vs. precision and Definitions

A key problem is the fact that provable security in cryptography requires that any algorithm only operate over numbers in a field. Thus, all numbers are required to be integral. This can cause havoc with an optimization algorithm, since many numbers are expected to be real. Indeed, many algorithms are notoriously sensitive to the quality of the data. Simply adding a few bits of precision and converting to integers is not likely to be sufficient to fix the problems. Indeed, this might require that the current definitions of security themselves be relaxed a little to allow efficient handling of real v/s. integer issues, as well as to handle the issue of malicious adversaries. Protecting against malicious adversaries currently requires great effort and causes severe efficiency constraints. However, given the subtle nature of security, the implications of any change must be carefully studied.

### 3.2.5 Game Theoretic approaches to malicious adversaries

Significant work is possible at the intersection of game theory and cryptography. Mechanism design (in game theory) and cryptography (in computer science) are both dual and opposites of each other. Cryptography has been concerned with allowing agents to hide information while achieving some particular objective (for example, jointly computing a function). Mechanism design has been concerned with forcing agents to reveal information while achieving a particular objective. This has many ramifications. For one thing, it explains why cryptography has managed for so long to avoid explicit modeling of the agents' utility functions. There is an opportunity to fuse the two perspectives and speak about informational mechanism design. IMD is characterized by the fact that each agent's utility is a function of the informational structure – who knows what. We need to figure out some way to explicitly model utilities. There has been some work at the intersection of cryptography and game theory. Fischer and Wright[29] provide an application of game theoretic techniques to the analysis of a class of multi-party cryptographic protocols for secret bit exchange.

Dodis et al.[23] provide a cryptographic protocol to the correlated element selection problem. Teague[79] extends this protocol to work also for non-uniform distribution. Other work that addresses the same problem without help from a third-party mediator includes [7, 11, 32, 46, 4]. Matsuura[55] provides a survey of the emerging interdisciplinary area between information security and economics. In [20], we explore secure solutions to the transportation load swapping problem, and show how such a protocol can be incentive compatible, thus protecting from malicious adversaries. This needs to be explored further and much more research is necessary.

### 3.2.6 Ensuring well formedness of the inputs

Another related issue is how to ensure that participants stick to their correct inputs for the protocol. Mechanism design can help with this as well ensuring that if one cheats, either they are caught or else they suffer. In [20] we show a representative example, where all parties are bound to submit their correct inputs as well as stick to the protocol due to their own incentives. However, in general this may be difficult. Other solutions may need to be explored.

### 3.2.7 Security definitions for difficult problems

One of the biggest successes of SMC is that it has provided a solid quantifiable mathematical way of proving the security of an algorithm. One can have confidence in the security of an algorithm after it has been proven secure in the SMC framework. However, the computational security definitions in SMC depend on the fact that problems can be solved in polynomial time (i.e., the solutions are “efficient”). What happens when we are considering problems that are exponential in the worst case? A new set of security definitions must be formulated to define and quantify security in this case. Optimization is a good candidate for this since many solutions for non-linear programming and integer programming are exponential in the worst case. We will explore this issue as we go along.

## 3.3 Privacy-Preserving Linear Programming

Linear Programming (LP) is an important sub-field of optimization where the objective function and the constraints are all linear. Linear Programming models are applicable to a wide variety of problems. Well known examples arise in many industries including transportation, commodities, airlines, communication, etc. There are also a variety of military applications and other economic applications. Software for linear programming (including network linear programming) consumes more computer cycles than software for all other kinds of optimization problems combined. LP is a very rich area of research with numerous algorithms known. Solution approaches from LP have also benefited the overall field of optimization.

Again, the main question arises when the constraints and/or the cost function are distributed in some fashion. For example, let us consider a specific subproblem – the data is distributed between two parties; the constraints with one while the cost function with the other party. Now, the two parties wish to find the optimal solution without revealing their private data to each other. Such a situation can often arise. For example, the military needs to have supplies shipped from manufacturing sites to various bases. For this, they would like to employ the services of a transportation provider. But rather than provide details to all providers (even if they possess security clearance), they would prefer to have to give the exact details only to the transporter with whom they have a contract. However, they still need to figure out *who* is the lowest cost provider. We now look at two possible approaches to solving this problem; we will also see how solutions provide techniques and insights that can be used to address other problems.

**A Transformation Approach** A possible solution is to transform the vector space by applying a linear transformation. This idea was first proposed by Du to solve systems of linear equations[25]. Du later proposed extending this idea to solve the two party linear programming problem[24]. However their solution sketch gives incorrect results in many cases (a non-optimal solution is incorrectly reported as optimal), and is limited to two parties. Still, the idea has merit and should be further explored. In fact, we now sketch a correct solution. Assume that you want to solve the problem:  $\min c^T x$  s.t.  $Mx \leq B, x \geq 0$ . The key to the solution is based on the fact that  $MQ^{-1}x \leq B$ , and  $Q^{-1}x \geq 0$  if  $Mx \leq B$  and  $x \geq 0$  (the elements of  $Q^{-1}$  should all be positive). Let  $M' = MQ$ ,  $y = Q^{-1}x$ , and  $c'^T = c^T Q$ . We now have a new linear programming problem:  $\min c'^T y$  s.t.  $M'y \leq B, y \geq 0$ . Du [24] correctly proves that if  $y^*$  is the solution to this problem,  $x^* = Qy^*$  must be the solution to the original problem that minimizes  $c^T x$ . The proof is based on contradiction.

	$P_1$ solves	$P_2$ solves
$P_1$ knows	(1)	(2)
$P_2$ knows	(3)	(4)

Table 1: The four possibilities

Even assuming that it is possible to secretly compute the transformed matrix, how does this help privacy? That depends on who knows the  $Q$  matrix and who solves the modified LP problem. Assume that  $P_1$  knows the cost function and  $P_2$  knows the constraints. Figure 1 shows the four possible combinations – based on who knows  $Q$  and who solves the new LP problem. Let us consider each of them in turn. In (1),  $P_1$  determines  $Q$  and solves the new LP problem. In this case,  $P_1$  knows a  $n \times n$  invertible matrix and learns the  $m \times n$  matrix  $MQ$ . It is an easy task to post-multiply by  $Q^{-1}$  and thus to retrieve  $M$ . This leads to no security for  $P_2$ . Similarly, in (4),  $P_2$  knows  $Q$  and solves the new LP problem. In this case,  $P_2$  can compute  $Q^{-1}$  and knowing  $C^T Q$ , can retrieve  $C^T$ . In this case, there is no security for  $P_1$ . That leaves us with (2) and (3). In (2),  $P_1$  knows  $Q$ , but  $P_2$  solves the modified LP problem. In this case,  $P_2$  gets  $MQ$  as well as  $C^T Q$ . Since  $M$  is a  $m \times n$  matrix it is non-invertible, thus it is not possible to exactly learn  $Q$ . However, that does still leave  $P_2$  with  $m$  linear equations in  $n$  unknowns for each column vector of the  $Q$  matrix. The most secure case is (3), where  $P_2$  knows  $Q$  and  $P_1$  solves the problem. In this case,  $P_1$  learns  $MQ$  as well as  $C^T Q$ . Knowing  $C^T$  enables it to learn only *one* linear equation in each of the column vectors of  $Q$ . There is no way for it to recreate  $M$  in any way. Is this the best possible? Not necessarily. There are two more alternatives.

The first is the possibility of using an untrusted third party.  $P_1$  and  $P_2$  jointly compute/decide on the matrix  $Q$ , and use it to create the modified LP problem. They now send this to the third party which actually solves the problem and returns the solution to them. As long as the third party does not collude with either  $P_1$  or  $P_2$  they are completely secure. This is the most secure alternative but it depends on the existence of an untrusted third party. The final alternative is the most interesting.  $P_1$  and  $P_2$  jointly create the  $Q$  matrix, but each holds a share of it, i.e.,  $Q = Q1 + Q2$ , with  $Q_i$  held by  $P_i$ . Now they can compute  $MQ$  as well as  $C^T Q$ , and either one can then go ahead and solve the problem. Let us now analyze what happens in either case. If  $P_2$  solves the LP problem.  $P_2$  gets  $MQ1 + MQ2$ . It can subtract  $MQ2$  to get  $MQ1$ . In this case,  $P_2$  knows  $m$  linear equations in  $n$  unknowns for each of the columns of  $Q1$  which is not sufficient to determine  $Q1$ .  $P_2$  will also find out  $C^T(Q1 + Q2)$ . But without knowing  $Q1$ , there is no way to determine  $C^T$ . The other case, where  $P_1$  solves the LP problem is better for security. Here  $P_1$  gets  $C^T Q1 + C^T Q2$ . Subtracting  $C^T Q1$ , it can learn  $C^T Q2$  but this only gives it one equation in  $n$  unknowns for each column of  $Q2$ , which is quite insufficient to determine the real values.  $P_1$  also learns  $MQ1 + MQ2$  but without knowing  $Q2$  there is no way for it to determine  $M$ . Thus, this is the better of the two options.

One might think that splitting the  $Q$  matrix gains us significant advantage in terms of security. But this is not quite true. Let us extend our analysis further. If we look at the best cases seen so far, the only difference is that  $P_1$  would learn  $M(Q1 + Q2)$  where it only knows  $Q1$  as against learning  $MQ$  where it knows nothing. It is not clear that this is really any better (in fact it might be worse). The final detail necessary to show the feasibility of the approach is a method to securely transform a matrix. This is possible using several different ways, including the method proposed by Du[24, 26] based on Oblivious Transfer.

The above discussion shows the feasibility of transformation for two parties with the specific data distribution. However, in general, several questions remain. First, is the fact that the elements of the  $C$  and  $M$  matrices could be real, where as the secure transformation procedure will only operate on integers (since Homomorphic Encryption operates over a closed field of integers). However, assuming a constant precision, it is easily possible to convert all of the numbers into integers. Also, since none of the operations involve division, we do not need to worry about truncation or rounding off errors. The question of security is also somewhat unclear. While the security of the method can be easily discussed, the question of what can be learned from the transformation itself is still unknown. For example, does knowing the transformed constraints reveal any useful information? While it is clear that the original constraints cannot be reconstituted from the transformed ones, what about other features. Can you infer anything about the hardness of the problem, or the type of constraints, or their relation to each other? All of these are significant questions that must be solved before the method could be used in practice. Some of the work in secure computation shows possible solutions approaches. If we could show that the results of any function computed from the constraints in polynomial time are indistinguishable from random numbers uniformly generated, we can prove

that transformation leaks nothing. This needs to be further explored. A final observation is that it should be possible to adapt the transformation approach to work for integer programming as well as quadratic programming. For integer programming the case is exactly the case as for linear programming. For quadratic programming, the form of the constraints changes – therefore the transformation method must take this into account. However, polynomial evaluation may be used to still perform the transformation.

### 3.3.1 A secure Revised Simplex method

We now look at a different tack to the problem: Taking a particular solution method and making a privacy-preserving variant of it. The Revised Simplex method is an efficient method based on some of the early theory developed in the field. Some of the most basic theorems in LP state that:

- The set of all feasible solutions to the LP problem is convex.
- The objective function assumes its minimum at an extreme point of the convex set  $K$  generated by the set of feasible solutions to the linear programming problem. If it assumes its minimum at more than one extreme point, then it takes on the same value for every convex combination of those particular points.
- If  $X = (x_1, x_2, \dots, x_n)$  is an extreme point of  $K$ , then the vectors associated with positive  $x_i$  form a linearly independent set. From this, it follows that at most  $m$  of the  $x_i$  are positive (the others are zero).

The proof of the above theorems can be found in any good book on Linear Programming (e.g. [31, 45, 9]...). Taken together, this means that one simply needs to search through at most  $\binom{n}{m}$  (n-choose-m) extreme point solutions. Since, for large  $n$  this is still an extremely high number, a computationally feasible technique needs to be found to appropriately go through solutions and iterate to the optimum. The *Simplex method* is just such a technique. The Simplex method first finds a basic feasible solution and then successively selects solutions which improve the optimum. Thus, in a finite number of iterations, the Simplex Method will converge to an optimum solution. For computer implementation, a more efficient matrix oriented approach known as the Revised Simplex method is used. The basic steps are as follows:

1. Determine the current basis.
2. Choose the variable to enter the basis based on the greatest cost contribution.
3. If the variable cannot decrease the cost, we have the optimal solution.
4. Determine the variable that exits the basis.
5. If the variable can increase without causing any other variable to leave the basis, the solution is unbounded.
6. Pivot on the appropriate element to get the new basis and repeat from Step 2.

Now we describe how to get a secure solution in our specific case; i.e., when the data is distributed between two parties with the first knowing the objective function and the second knowing the constraints. The key is to realize that only steps 2 and 3 require interaction. Since the second party knows the entire constraint matrix it can determine steps 1, 4, 5, and 6 on its own. To determine the variable to enter the matrix, one requires  $n$  secure scalar products followed by  $n$  secure comparisons. We already know how to securely compute the scalar product ([33]) and how to securely compare ([89]). Thus, these are used as components in the complete solution. An interesting point is to note that the secure scalar product and comparisons need to be carried out for every iteration. Thus, the overall running time of the algorithm depends on the number of iterations required to solve the problem. One problem with the secure revised complex method is that in each iteration, the current feasible solution is revealed. Thus, the overall process reveals a string of feasible solutions leading to the optimal solution. How much information this reveals is an open question. Given the optimal solution, one may construct a cost function for which that solution would be optimal[3]. This could be used to get a possible path to the optimal solution, but not necessarily the same path. In fact, different cost functions might easily give different paths to the solution. Thus, revealing the path to the optimal solution does have some security implications. Research is necessary to determine the amount of information leakage.

### 3.3.2 Extension to multiple parties

The above two sections explored the challenge of distributed linear programming for two parties. However, the solutions formulated are specific to a very restricted problem. In the case of the secure revised simplex method, much of the solution is tied to the specific way in which the data is distributed. In the general case, where data is arbitrarily distributed, the problem becomes significantly more difficult. Similarly, the security and feasibility of the transformation approach need to be explored for different data distributions. With more than two parties, the problem becomes even more complex. Further more, large problems cause significant resource constraints. Radical solution approaches might be necessary to efficiently solve the problem. We then look towards current distributed solutions in the hope that they may lead us towards an efficient solution and then strive to make it secure. For example, Dantzig-Wolfe Decomposition is often applied to problems having block angular structure. Block angular systems consist of some common rows which contain nearly all the variables. The remaining rows are divided into sets with corresponding variables that are contained in the rows. Each variable is in exactly one set. These sets define subproblems. This occurs very often in large structured problems. In the Dantzig-Wolfe decomposition, we create one master problem and several subproblems. The master problem will handle the common constraints by asking the subproblems for proposals. The master problem will choose a combination of proposals that maximizes profits while meeting the common constraints. The advantages of this sort of decomposition are: the subproblems can be solved independently; at no time is a large l.p. solved, only ones as big as the subproblems plus the master problem; the central controller does not need to get into details on how the proposals are generated. It is enough that they can be for any cost function; if the subproblems have special structure (e.g., perhaps one is a transportation problem) then those specialized solution techniques can be used. This also makes it easier to preserve privacy if the large problem could be solved without knowing the precise solutions of the sub-problems. Practically, the main drawback of this approach is in possible convergence problems. Normally, this method gets very good answers quickly, but it requires a lot of time to find the optimal solution. The subproblems may continue to generate proposals only slightly better than the ones before. Thus, we might have to stop with a sub-optimal solution for efficiency reasons. In general there are other distributed solutions that have also been developed to deal with the case of incomplete information, or to reduce the overall communication. These can also be adapted and made secure.

### 3.4 Research Plan Summary

Sections 3.1 and 3.2 have shown specific problems and challenges that will be addressed as the project goes forward, more details are given in Section 5. Section 3.3 demonstrates particulars of the challenges, and approaches that will be taken to address them. These are broad outlines and specific tools will be built for each specific problem area. An overall system for distributed optimization will have to be developed. As much as possible, developed tools will be integrated with existing systems to maximize use (for example, the Linear Programming modules should be integrated with the GNU Linear Programming Kit[54]).

## 4 Education Plan

This research also lays the foundation for the education plan. The education plan has three key objectives: 1) increase undergraduate interest in privacy, security, and data analysis 2) introduce MBA students to the concepts of secure computation 3) spark widespread research in this area at the graduate level.

To meet these objectives, the PI will develop a new curriculum that presents the value and feasibility of secure optimization as a part of secure information sharing. Basic privacy/security concepts will be taught at the undergraduate level. Motivation is necessary at every level, but especially so for undergraduates. The use of real examples and actual business problems should spark widespread interest in the field. A stand-alone course on privacy, secure computation, and optimization will be developed and offered as a seminar course to graduate students. Deep insight into methods are expected at the Ph.D. level. Additionally, one of the biggest impacts the project will have is at the MBA level. With MBAs, we intend to explore the value of information sharing at the managerial level. Simply making them aware of the possibility of secure computation, and pointing them in the right direction should be a real gain. Overall, by exposing MBA students to these concepts, we expect further penetration of these theoretical concepts into *real* use.

These objectives are consistent with the mission of the PI's department, and also contribute to addressing the nation's need by developing a qualified work force with security knowledge and skills. The Management

Science and Information Systems Department at Rutgers University has identified information systems security as one of the two areas to strengthen (along with Supply Chain Management) in the near future. As one of the faculty who is responsible for security, data mining, and database research and education, the PI plans to contribute significantly to the department's overall goal, and build up his career on the basis of quality research and education in the field of secure data analysis and information sharing.

Though privacy and secure computation has been studied actively in the research community, to our best knowledge, there are currently no systematic course materials on secure optimization and information sharing; indeed, the field itself is quite new. There are many courses in secure computation, but these emphasize theoretical concepts for the most part. Our goal is to introduce theoretical concepts, while providing significant motivation through applications, thus sparking further research. Development of case studies for the MBA courses will be crucial towards integrating this into real use. The proposed educational activities are aimed at developing portable course materials and this will significantly reduce the time and effort required to teach secure information sharing in undergraduate or graduate courses. The proposed educational activities will be integrated with the proposed research, benefiting from research on techniques and tools, and contributing to research in training and attracting students to validate and potentially improve the proposed techniques. The portability of the course materials will be emphasized to facilitate the adoption or adaptation of these course materials by other faculty or institutions. One of my goals is to also write a book on the subject of secure optimization with sufficient case studies and motivation problems. Being one of the pioneers in Privacy-Preserving Data Mining, I have already co-authored a monograph on the subject [87] for the Advances in Information Security bookseries by Springer-Verlag. I expect to work on such a book for secure optimization through the project, especially during my sabbatical.

#### 4.1 Prior Educational Accomplishments

The PI has been actively involved in teaching, advising, and curriculum development activities since joining the Management Science and Information Systems Department at Rutgers University, and has made initial contributions in teaching, curriculum development, and student advising. The PI was a member of the curriculum revision/development committee for the Undergraduate MSIS major at the New Brunswick Campus as well as the Undergraduate MIS major at the Newark Campus. This has led to the introduction of new topics and revitalization of the major. In fall '05 and fall '06, the PI developed and taught the undergraduate security elective and the undergraduate introductory programming core course from scratch. This was quite successful, and led to an independent study with one of the students (Artem Adamov) on secure computation issues. The research done here has culminated in two journal articles (in preparation). In the MBA Computer & Information Systems course, students were exposed to the idea of secure computation and groups were asked to prepare hypothetical case studies where secure computation would be of interest. This led to very interesting discussions and sparked interest in the area. The PI has also been on the advising committee of three Ph.D. students and is currently co-advising three students for their dissertation research.

#### 4.2 Proposed Activities

**Course Development** Secure information sharing is an important topic for a graduate course for the following reasons: (1) privacy and security considerations are critical to any data collection and analysis (2) data analysis and use is necessary to retain competitive advantage; (3) secure computation has a rich theory of its own, developed over the last 35 years; and (4) secure computation is an active research field.

The PI will develop portable course modules on data privacy, optimization and secure information sharing. After evaluation, these will be shared with other higher educational institutions. The first module will consist of secure linear programming techniques proposed in the first research component. The second module will consist of various secure techniques proposed by other researchers for the constraint satisfaction problem [73, 77, 92, 74] and incorporate the security analysis techniques developed in the second research component. For each module, the PI will include critical techniques that bring in new capabilities. This will enable the PI to develop and evolve a course on secure information sharing, with the objective of having students gain deep understanding of the power and limitations of secure computation as also the feasibility of it. The PI will develop a number of lab exercises and projects to give students hands-on experiences of what they learn in classrooms. This will help integrate the proposed research and educational activities. These lab exercises will allow the students to examine the effectiveness, the efficiency, and the limitations of critical techniques taught in class. The PI will associate open questions with lab exercises to encourage the students

to get involved in research. As deliverables of the proposed educational activities, each course module will consist of a description of prerequisites, reading assignments, slides for classroom instructions, homework assignments, lab exercises, and related research topics. The proposed research will provide materials in all these parts, especially the tools and test cases for lab exercises and research topics. The course materials will be disseminated through web-site, educational conferences, and personal communication with faculty in other institutions.

For evaluation, the PI plans to initially incorporate these modules in the courses he currently teaches (e.g., Information System Security). The PI will appropriately adjust the scope and depth of these course modules and incorporate them into other related courses. The PI will also develop a stand-alone course on secure information sharing and privacy and request feedback from the students in class and instructors from other faculty by developing an on-line form with questions specifically targeted at the effectiveness of the new course modules. The PI is also collaborating with Professor Soon Chun from CUNY College of Staten Island on developing an integrated IAS curriculum. She is happy to use the course materials developed in this project and further disseminate them through the CUNY system<sup>1</sup>. 16 out of 19 CUNY colleges are minority serving institutions. Rutgers also has a very diverse student body. The U.S. News and World Report has named Rutgers-Newark the most diverse university in the country for six consecutive years. Since privacy is of particular interest to some underrepresented groups (especially, to people from lower socio-economic groups[70]), we anticipate this will have particular impact on grooming promising undergraduates from such groups for Computer Science and Information Systems research careers.

## 5 Agenda and Deliverables

The ultimate goal of the work proposed here is to produce a prototype system for privacy-preserving optimization which can fully conduct optimization analysis and satisfy organizational privacy and confidentiality specifications. Apart from building general solutions, a suite of tools will be developed that provide efficient solutions for specific sub-problems. This system will be made available to the academic community as well as industry so other researchers can benefit from their use. We are also working with the Customer Relationship Management (CRM) Center at Rutgers. The CRM center has been extremely successful in forging research-industry partnerships, and getting access to real-world test data. Partnering with the CRM center we intend to test our algorithms against real data sets and formulate more real problems. Through discussion with companies, we will identify and work on problems that are significant to them. From a broad industrial perspective, the tools developed should herald a paradigm shift in the way business is conducted.

### Project Timeline:

- **Year 1:** The proposed transformation based approach for Linear Programming will be further refined and methods developed that apply over other possible data distributions. The PI will develop the first portable course module, and tackle the fundamental challenge of utility metrics for transformation.
- **Year 2:** Distributed techniques for linear programming such as Dantzig-Wolfe will be developed, implemented and evaluated. Methods for quadratic programming will be developed, and efficient solutions developed for iteration. The challenge of modeling exterior knowledge and result analysis will be tackled. The second portable course module will be developed.
- **Year 3:** Prior work on distributed constraint satisfaction will be evaluated according to secure multi-party computation criteria and methodology and new techniques developed. Comprehensive performance evaluation using large real data sets will be conducted. The course modules will be incorporated in existing courses for feedback. The challenge of mechanism design will be met.
- **Year 4:** A preliminary version of the secure optimization system, albeit with limited capabilities, will be available. A complete evaluation using real data sets will be conducted. Feedback from users will be used to refine the prototype system. The first stand-alone course regarding secure information sharing and optimization will be developed and offered as a seminar course to graduate students. The survey of CIOs will be carried out to gauge penetration of the work into the real world.
- **Year 5:** The methods will be fine tuned and the prototype system will be disseminated to all related communities. The secure information sharing and optimization course will be offered again. It is expected that the course will be much improved based on the first offering and domain users' feedback.

---

<sup>1</sup>See the support letter from Professor Soon Chun

## 6 Success Metrics

What measures of success are applicable to this work? The key characteristics to consider are the security of proposed solutions, as well as the suitability for optimization analysis. A good solution for privacy preserving optimization analysis must meet several criteria:

**Quality of results.** When measuring how well our methods performs for a specific optimization technique, we need to measure the quality of results obtained. An obvious comparison point is the actual result obtained from running the specific algorithm over the original data unrestricted by security concerns. While achieving exactly the same results may be ideal, at times, approximate solutions might be much more efficient and secure. Thus, we will measure how well the secure technique performs over data as against the standard technique it is designed to replace. The secure method will be judged good for the specific problem domain, as long as the results are sufficiently good – i.e., within a bounded error of operating over the centralized data. For example, we may be able to show that our secure linear programming method is guaranteed to return a feasible solution within 10% of the optimum, but not necessarily the optimum itself. A more difficult measure is the effectiveness of results for guiding further analysis. This is an open challenge; we will explore this issue as part of the project.

**Applicability to generic optimization.** Optimization has many important sub-fields. A class of solutions might work for one field but not for others. We need to evaluate how generic are the developed solutions. Can multiple analyses be run without compromising on the quality of results? While testing a method against every possible optimization technique is clearly infeasible, it should be possible to figure out what classes of techniques work appropriately with the method. We will compare our methods according to how many and what kind of optimization analyses do they support. Thus, we might be able to make recommendations for specific categories like linear programming, quadratic programming, etc.

**Computational cost.** Standard computational measures, such as worst-case running time, are often inappropriate for optimization. Algorithms that work well in practice may have intractable worst-case running time or space requirements, but data causing worst-case performance may be rare or nonexistent in practice. For example, the revised simplex method for linear programming is exponential in the worst case for many standard pivot rules. However, under smoothed analysis[76], the running time is polynomial. In any case, we do not simply worry about the cost of running the optimization technique – what we are interested in is the overall cost of transformation for any generalized solution. In such a case, standard measures may suffice, especially in terms of comparing the multiple competing algorithms in the absolute sense. We will however measure the cost of the secure algorithm relative to the algorithm without the constraint of security, e.g., “ $O(\log k)$ \*transformation running cost, where  $k$  is chosen depending on the amount of privacy required”.

**Communication cost.** The two standard measures used in practice to measure the communication cost are 1) the total number of bits communicated, and 2) the number of messages exchanged. We will determine bounds in terms of the computational measures of centralized algorithms performing the same task.

**Security measures.** Perfect security is rarely possible. Indeed, if any meaningful computation is to be carried out, the results themselves reveal some amount of information. Other problems can occur – Encryption can be broken, outside knowledge added to information passed by the algorithms may reveal individual values, etc. Rather than discuss security on a per technique basis, we would prefer to talk about security on the dataset scale. Thus, we would like to quantify exactly what information about the original dataset is revealed through the final solution and the algorithm. For example, we might be able to restrict the scope of possible cost functions that would give such a result, but reveal no other information. Any information inferable from this is also automatically revealed, though not necessarily in an explicit fashion.

Another key measure of success of this research will be the ease of addressing a new real-world problem: Will these efforts require considerable efforts or will we succeed in developing a methodology enabling new problems to be addressed as undergraduate research projects? This will largely be measured through outreach to undergraduate and masters students both in the classroom and through research. In particular, we plan to seek Research Experiences for Undergraduates supplemental funding to explore our ability to make this technology widely accessible. Overall, the success of this project will result in fundamental advances in research and have huge educational, economic and societal impacts.